UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/553,721	09/06/2006	Lionel Wolovitz	307742US91PCT	2053

22850 7590 04/15/2009

OBLON, SPIVAK, MCCLELLAND MAIER & NEUSTADT, P.C.
1940 DUKE STREET
ALEXANDRIA, VA 22314

EXAMINER	
RUBIN, BLAKE J	

ART UNIT	PAPER NUMBER
2457	

NOTIFICATION DATE	DELIVERY MODE
04/15/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patentdocket@oblon.com
oblonpat@oblon.com
jgardner@oblon.com

Office Action Summary	Application No.	Applicant(s)	
	10/553,721	WOLOVITZ ET AL.	
	Examiner	Art Unit	
	BLAKE RUBIN	2457	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *10 March 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-37* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-37* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☒ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is a response to a request to continued communication filed March 10, 2009.

2. Claims 1-37 are pending in this application. Claims 1, 34, and 36-37 are currently amended.

3. The application is a 371 of PCT/GB04/01685 filed April 19, 2004, which further claims foreign priority to United Kingdom patent application priority # 0308991.9, filed April 17, 2003.

Claim Rejections - 35 USC § 103

31. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-16, 19, 26, and 32-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Logston et al (PCT International Publication No. WO 01/77815 A2, hereinafter Logston), in view of Johnson et al (U.S. Patent Application Publication No. 2002/0049841, hereinafter Johnson), in further view of Piskiel et al (PCT International Publication No. WO 97/46939, hereinafter Piskiel).

32. With respect to claim 1, Logston discloses a data access, replication or communications system (page 1, lines 16-19) comprising:

A terminal (page 2, lines 1-3, *client devices*) including an electronic memory storing a terminal-side executable (page 1, lines 32-34, *DACP*) and a processor provided to execute the terminal-side executable (page 8, line 9) to enable a communication therewith (page 8, lines 27-28),; and

A server (page 2, lines 25-26) including an electronic memory storing a server-side executable (page 1, lines 32-34, *DASP*) and a processor provided to execute the server-side executable (page 8, lines 12-13) to enable a communication therewith (page 8, lines 27-28),

Wherein, the terminal-side executable and the server-side executable exchange messages using a message queuing system (page 19, lines 23-25) over a network (page 1, line 23) and cooperatively function as a client of a second server (page 2, lines 25-26).\

But does not disclose terminal-side and server-side message packet encapsulation and sizing.

However, Johnson discloses the server-side executable (paragraph [0047], lines 11-15) and terminal-side executable (paragraph [0046], lines 6-11) dividing a message into a plurality of packets (paragraph [0073], lines 7-13, *data reordering*), each packet having a size corresponding to a transport protocol payload size (paragraph [0137], lines 10-13),

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston with the packetization of Johnson. The motivation to do so being, to provide an efficient means for processing data across a computer network by accelerating the content delivery of the system (Johnson: paragraph [0137]).

The combination of Logston and Johnson does not disclose exchanging communication independent of a session-based protocol.

However Piskiel discloses exchanging communication independent of a session-based protocol (page 32, lines 15-20, *communication error…recover automatically*).

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston and Johnson with the connection independence of Piskiel. The motivation to do so being, to provide a robust means of communicating data across a network by maintaining synchronization between messaging queues regardless of any specific connection parameters.

33. With respect to claim 2, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1 wherein the message queuing system is message oriented middleware (page 15, lines 32-33).

34. With respect to claim 3, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Piskiel further discloses insulation of a terminal program (page 4, lines 14-18 and 24-27 where the terminal is disclosed as the *receiving*

node) from being affected (page 6, lines 12-13, *assure synchronization*) if a connection over the network is broken (page 32, lines 15-17, *communication error*) by queuing messages in readiness for the connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*), enabling the terminal program to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*).

35. With respect to claim 4, the combination of Logston, Johnson and Piskiel discloses the system of claim 1, Piskiel further discloses insulation of a server program (page 4, lines 14-18, where the server is disclosed as the *originating node*) from being affected (page 6, lines 12-13, *assure synchronization*) if a connection over the network is broken (page 32, lines 15-17, *communication error*) by queuing messages in readiness for the connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*), enabling the server program to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*).

36. With respect to claim 5, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Piskiel further discloses each message that is queued defines part or all of an event (page 4, lines 20-24, *associated transaction*), the event describing a change to data stored at either the terminal or server (page 11, lines 4-6) in enough detail to enable data replication to take place without a need for a synchronization engine (page 7, lines 9-15, *"exactly once"*), data replication being

achieved by sending events rather than a complete dataset (or sub-sets of a dataset) of stored data for synchronization (page 8, lines 2-8, *atomic operation*).

37. With respect to claim 6, the combination of Logston, Johnson, and Piskiel disclose the system of claim 5, Piskiel further discloses the terminal-side executable can create amd queue events (page 12, lines 26-28), enabling the terminal-side executable to proceed to another task (page 14, lines 26-27), even if a network connection is broken (page 32, lines 15-17, *communication error*). And Logston further discloses the events being queued in one of the terminal-side executable and a message queuing system (page 19, lines 23-25).

38. With respect to claim 7, the combination of Logston, Johnson, and Piskiel disclose the system of claim 5, Piskiel further discloses the server-side executable can create and queue events (page 12, lines 26-28), enabling the server-side executable to proceed to another task (page 14, lines 26-27), even if a network connection is broken (page 32, lines 15-17, *communication error*). And Logston further discloses the events being queued in one of the server-side executable and a message queuing system (page 19, lines 23-25).

39. With respect to claim 8, the combination of Logston, Johnson, and Piskiel disclose the system of claim 6, Piskiel further discloses the queued events persist in non-volatile memory when the terminal is switched off (page 33, lines 1-4).

40. With respect to claim 9, the combination of Logston, Johnson, and Piskiel disclose the system of claim 7, Piskiel further discloses the queued events persist in non-volatile memory when the server is switched off (page 33, lines 1-4).

41. With respect to claim 10, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the terminal-side executable and the server-side executable cooperatively function as middleware (page 1, lines 31-34, where middleware is anticipated by the *distributed application*) between a terminal program (page 1, lines 31-34, *DACP*) running on a wireless terminal (page 2, lines 3-4) and a server program (page 1, lines 31-34, *DASP*) running on the server (page 1, lines 31-34).

42. With respect to claim 11, the combination of Logston, Johnson, and Piskiel disclose the system of claim 6, Piskiel further discloses the messages queued on the terminal side are references to data stored on the server (page 13, lines 1-17, *updating of information stored*).

43. With respect to claim 12, the combination of Logston, Johnson, and Piskiel discloses the system of claim 10, Piskiel further discloses a message queuing system (page 4, lines 16-18, *wraparound queue*) on the terminal side (page 4, lines 18-20, *receiving node*) insulates the terminal program from being affected (page 6, lines 12-13,

assure synchronization) if a connection over the network is re-established (page 32, lines 24-25, *communication failure…is corrected*) by automatically causing a next message in a terminal-side queue to be sent (page 32, lines 24-27, *automatically recover the state of message processing*).

44. With respect to claim 13, the combination of Logston, Johnson, and Piskiel discloses the system of claim 10, Piskiel further discloses a message queuing system (page 4, lines 16-18, *wraparound queue*) on the server side (page 4, lines 16-18, *originating node*) insulates the server program from being affected (page 6, lines 12-13, *assure synchronization*) if a connection over the network is re-established (page 32, lines 24-25, *communication failure…is corrected*) by automatically causing a next message in a server-side queue to be sent (page 32, lines 24-27, *automatically recover the state of message processing*).

45. With respect to claim 14, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the terminal-side executable processes events from a terminal program (page 8, lines 32-33, *software program*), which is an e-mail or PIM program (page 9, line 3).

46. With respect to claim 15, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the server-side

executable processes events from a server program (page 9, lines 31-33), which is a mail server program (page 9, line 3).

47. With respect to claim 16, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the terminal is a wireless terminal (page 2, lines 3-4) such as a mobile telephone or smartphone (page 2, lines 3-7, *cellular telephones and personal digital assistants*).

48. With respect to claim 19, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1 wherein the server-side executable stores data on the server (page 12, lines 19-23, *downloads*) to assemble a sent message (page 12, lines 19-23, *to be subsequently propagated…copied*), to which fewer than all data was received from the terminal (page 12, lines 26-30, *subsequent downloads*).

49. With respect to claim 26, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the client, implemented by the terminal-side and server-side executables, enables a correct routing (page 10, lines 23-24) of messages addressed to a terminal identified by an ID (page 23, lines 7-9, *identifier*) by mapping that ID to an actual IP address needed to reach the terminal (page 23, lines 10-15).

50. With respect to claim 32, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, Logston further discloses wherein the client, implemented by the terminal-side and server-side executables, includes a distributed application platform (page 7, lines 19-21) that makes calls ([age 2, lines 32-34, *used on the network*) to a distributed communications platform (page 2, lines 29-34, *the given network*).

51. With respect to claim 33, the combination of Logston, Johnson, and Piskiel discloses the system of claim 32, Logston further discloses wherein the distributed communications platform enables reliable delivery of a message (page 2, lines 23-25) over the network (page 17, lines 15-18), independently of any unreliable transport protocol used (page 17, lines 15-18, *UDP*).

52. With respect to claim 34, Logston discloses a method of data access, replication or communication (page 1, lines 16-19) comprising the steps of:

running a terminal side executable (page 1, lines 32-34, *DACP*) on a terminal (page 2, lines 1-3, *client devices*) to enable communication with a server (page 2, lines 25-26) independent of a session-based protocol (page 1, lines 29-31), the terminal-side executable dividing a message into a plurality of packets, each packet having a size corresponding to a transport protocol payload size;

running a server-side executable (page 1, lines 32-34, *DASP*) on the server to enable communication with a terminal (page 8, lines 27-28) independent of a session-

based protocol, the server-side executable dividing a message into a plurality of packets, each packet having a size corresponding to a transport protocol payload size, there terminal-side executable and the server-side executable cooperatively functioning as a client to a second server (page 2, lines 25-26);

sending messages between the terminal-side executable and the server-side executable (page 8, lines 27-28) using a message queuing system (page 19, lines 23-25) over a networks (page 1, line 23).

But does not disclose terminal-side and server-side message packet encapsulation and sizing.

However, Johnson discloses the server-side executable (paragraph [0047], lines 11-15) and terminal-side executable (paragraph [0046], lines 6-11) dividing a message into a plurality of packets (paragraph [0073], lines 7-13, *data reordering*), each packet having a size corresponding to a transport protocol payload size (paragraph [0137], lines 10-13),

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston with the packetization of Johnson. The motivation to do so being, to provide an efficient means for processing data across a computer network by accelerating the content delivery of the system (Johnson: paragraph [0137]).

53. With respect to claim 35, the combination of Logston, Johnson, and Piskiel disclose the method of Claim 34, Logston further discloses wherein

the message queuing system is message oriented middleware (page 15, lines 32-33),

And Piskiel further discloses the terminal-side executable insulates the terminal from being affected (page 6, lines 12-13, *assure synchronization*) by a broken network connection (page 32, lines 15-17, *communication error*), enabling the terminal to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*), the terminal-side executable queuing messages in readiness for the network connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*), and

the server-side executable insulates the server from being affected by a broken network connection (page 32, lines 15-17, *communication error*), enabling the server to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*), the server-side executable queuing messages in readiness for the network connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*).

54. With respect to claim 36, Logston discloses a terminal (page 2, lines 1-3, *client devices*) comprising:

an electronic memory configured to store a terminal-side executable (page 1, lines 32-34, *DACP*) to enable communication therewith (page 8, lines 27-28);

a processor configured to execute the terminal-side executable (page 8, line 9),

But does not disclose packet encapsulation and sizing.

However, Johnson discloses dividing a message into a plurality of packets (paragraph [0073], lines 7-13, *data reordering*), each packet having a size corresponding to a transport protocol payload size (paragraph [0137], lines 10-13),

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston with the packetization of Johnson. The motivation to do so being, to provide an efficient means for processing data across a computer network by accelerating the content delivery of the system (Johnson: paragraph [0137]).

The combination of Johnson and Logston does not disclose exchanging communication messages independent of a session-based protocol, insulating against a broken connection, and re-establishing a connection including a queue automatically.

However Piskiel discloses exchanging communication independent of a session-based protocol (page 32, lines 15-20, *communication error…recover automatically*).

to insulate the terminal from a broken network connection (page 32, lines 15-17, *communication error*), enabling the terminal to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*),

queue messages in readiness for the network connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*), the messages being at least a part of an event describing a change to data (page 4, lines 20-24, *associated transaction*) stored on a server (page 11, lines 4-6) and allowing data replication without transmission of an entire dataset (page 8, lines 2-8, *atomic operation*), and

automatically send the queued messages upon re-establishment of the network connection (page 32, lines 20-24, *state of the message queue is therefor retained*) via at least one of a wired connection or a wireless connection to the network (page 10, lines 10-12).

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston and Johnson with the connection independence of Piskiel. The motivation to do so being, to provide a robust means of communicating data across a network by maintaining synchronization between messaging queues regardless of any specific connection parameters.

55. With respect to claim 37, a server (page 2, lines 25-26) comprising:

an electronic memory configured to store a server-side executable (page 1, lines 32-34, *DACP*) to enable communication therewith (page 8, lines 27-28);

a processor configured to execute the server-side executable (page 8, lines 12-14)

divide a message into a plurality of packets, each packet having a size corresponding to a transport protocol payload size,

But does not disclose message packet encapsulation and sizing.

However, Johnson discloses dividing a message into a plurality of packets (paragraph [0073], lines 7-13, *data reordering*), each packet having a size corresponding to a transport protocol payload size (paragraph [0137], lines 10-13),

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston with the packetization of Johnson. The motivation to do so being, to provide an efficient means for processing data across a computer network by accelerating the content delivery of the system (Johnson: paragraph [0137]).

The combination of Logston and Johnson does not disclose exchanging communication independent of a session-based protocol, insulating against a broken connection, and re-establishing a connection including a queue automatically.

However Piskiel discloses exchanging communication independent of a session-based protocol (page 32, lines 15-20, *communication error…recover automatically*).

to insulate the terminal from a broken network connection (page 32, lines 15-17, *communication error*), enabling the server to proceed to another task (page 32, lines 24-27, *automatically recover the state of the message processing*),

queue messages in readiness for the network connection to be re-established (page 32, lines 20-24, *state of the message queue is therefor retained*), the messages being at least a part of an event describing a change to data (page 4, lines 20-24, *associated transaction*) stored on a terminal (Logston: page 2, lines 1-3, *client devices*) and allowing data replication without transmission of an entire dataset (page 8, lines 2-8, *atomic operation*), and

automatically sending the queued messages upon re-establishment of the network connection (page 32, lines 20-24, *state of the message queue is therefor*

retained) via at least one of a wired connection or a wireless connection to the network (page 10, lines 10-12).

It would have been obvious to one skilled in the art at the time the invention was made to combine the distributed architecture of Logston and Johnson with the connection independence of Piskiel. The motivation to do so being, to provide a robust means of communicating data across a network by maintaining synchronization between messaging queues regardless of any specific connection parameters.

56. **Claim 17 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 1 above, in view of Hutcheson et al (U.S. Patent No. 6,947,761, hereinafter Hutcheson).**

57. With respect to claim 17, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, but does not disclose GPRS or UMTS.

58. However, Hutcheson discloses the network is a wireless WAN network (column 7, lines 24-25) such as a GPRS (column 7, lines 19-20) or UMTS network (column 7, lines 22-23).

59. It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the wireless networks of Hutcheson. The motivation to do so being, to provide a more versatile means of communicating over the network by having the ability to connect of preexisting wireless networks.

60. **Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 1 above, in view of Suarez (U.S. Patent No. 5,790,789).**

61. With respect to claim 18, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, but does not disclose a logon password.

62. However, Suarez discloses the server-side executable stores (column 15, lines 58-60, *database stored procedure*) a logon password (column 16, line 32) sent from the terminal-side executable (column 16, lines 10-13, *at least one agent*) and can use the logon password to access a server program (column 16, lines 36-43, *launching the service*).

63. It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the authentication of Suarez. The motivation to do so being, to provide a more secure means of communicating over the network by having the ability to authenticate users.

64. **Claims 20-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 1 above, in view of De Mendonca et al (U.S. Patent Application Publication No. 2004/0172453 , hereinafter Mendonca).**

65. With respect to claim 20, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, but does not disclose automatically deleting data.

66. However, Mendonca discloses the terminal-side executable monitors available memory on the terminal (paragraph [0005]) and automatically deletes data stored on the terminal (paragraph [0016], lines 4-9, where data is disclosed as *e-mail body text*) that meets pre-defined criteria of at least one of age, use, and size (paragraph [0016], lines 4-9, *usage based rules or time schedules*) without affecting a corresponding data stored on the server (paragraph [0016], lines 4-9).

67. It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the data archiving of Mendonca. The motivation to do so being, to provide a more resourceful means of communicating over the network by purging extraneous data stores.

68. With respect to claim 21, the combination of Logston, Johnson, Piskiel and Mendonca disclose the system of claim 20, Mendonca further discloses a user option to delete data stored on the terminal without affecting the corresponding data stored on the server (paragraph [0016], lines 4-9) is displayed at a same level in a menu hierarchy displayed on the terminal as an option to delete data stored on the terminal (paragraph [0017], lines 7-10) together with the corresponding data stored on the server (paragraph [0017], lines 11-15, *delete from the mail server*).

69. With respect to claim 22, the combination of Logston, Johnson, Piskiel and Mendonca disclose the system of claim 20, Mendonca further discloses the data is message data and the terminal side executable retains data that allows the message data to be re-supplied from the server if requested by a user (paragraph [0016], lines 4-15, *header information*).

70. With respect to claim 23, the combination of Logston, Johnson, Piskiel and Mendonca disclose the system of claim 20, Mendonca further discloses the data is not released from memory if the data is marked as unread, open for user viewing or action, or there is a pending action related to the data requesting additional data from the second server (paragraph [0016], lines 4-15, where the data being marked as open for action is disclosed as *header information*).

71. With respect to claim 24, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, but does not disclose various formats.

However, Mendonca disclose the terminal-side executable enables a document attachment (paragraph [0016], lines 13-14) to be sent to the wireless terminal (paragraph [0002], lines 1-3, *wireless information devices*) in either an original format in which the document is stored on the server (paragraph [0016], lines 4-9, where the original format is disclosed by the e-mail prior to the *body text…[*being*]…deleted*) or in another usable format converted from the original format (paragraph [0016], lines 4-9,

where the converted format is disclosed by the e-mail after the *body text…[*being*]…deleted*).

It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the format compatibility of Mendonca. The motivation to do so being, to provide a more resourceful means of communicating over the network by enabling various formats of data to be maintained which require different amounts of storage on different devices.

72. With respect to claim 25, the combination of Logston, Johnson, and Piskiel discloses the system of claim 1, but does not disclose deleting messages.

However, Mendonca discloses the terminal-side executable enables a user to select a release option to delete a message stored on the terminal without deleting a corresponding message stored on the server (paragraph [0016], lines 4-9) and to select a delete option to delete a message stored on the terminal (paragraph [0016], lines 4-9) and the corresponding message on the server (paragraph [0017], lines 4-15, *delete from mail server*), the release and delete options appearing at a same level in a menu hierarchy displayed on the terminal (paragraph [0017], lines 7-11).

It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the organizing interface of Mendonca. The motivation to do so being, to provide a more resourceful means of communicating over the network by enabling storage by various devices to be controlled though a menu.

73. **Claims 27 and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 26 above, in view of Guruprasad (U.S. Patent No. 6,802,068).**

74. With respect to claim 27, the combination of Logston, Johnson, and Piskiel discloses the system of claim 26, but does not disclose a NAT box.

75. However, Guruprasad discloses the address is a dynamic IP address (column 3, line 20) allocated by a NAT box (column 13, lines 50-52).

76. It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the Network Address Translator of Guruprasad. The motivation to do so being, to provide a more versatile means of communicating over the network by allowing flexibility in assigning terminal addresses.

77. With respect to claim 28, the combination of Logston, Johnson, Piskiel, and Guruprasad discloses the system of claim 27, Guruprasad further discloses the client, implemented by the terminal-side and server side executables, only initiates a message transfer if there exists a valid mapping (column 15, lines 39-42, *QoS considerations*).

78. **Claim 29 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, Piskiel, and Guruprased, as applied to claim**

28 above, in view of Iyer et al (U.S. Patent Application Publication No. 2004/0203749, hereinafter Iyer).

79. With respect to claim 29 the combination of Logston, Johnson, Piskiel and Guruprasad discloses the system of claim 28, but does not disclose refreshed mapping.

However, Iyer discloses wherein a mapping is refreshed (paragraph [0050], lines 18-23) whenever a specific kind of dedicated message having fewer than 20 bytes is received from the terminal (paragraph [0020], lines 18-23, *"keep-alive" message*).

It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, Piskiel, and Guruprasad with the remapping of Iyer. The motivation to do so being, to provide a more efficient means of communicating over the network by allowing the status of network to be updated periodically using minimal network capacity.

80. **Claim 30 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 1 above, in view of Podgorny et al (U.S. Patent No. 6,078,948, hereinafter Podgorny).**

81. With respect to claim 30, the combination of Logsto, Johnson,n and Piskiel discloses the system of claim 1, but does not disclose locking an application.

a. However, Podgorny discloses the terminal-side executable allows a server administrator (column 15, lines 66-67) to lock an application on the terminal

(column 13, lines 54-60) without affecting other applications on the terminal (column 7, lines 65-67, *implementation specific*).

It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the security of Podgorny. The motivation to do so being, to provide a more secure means of communicating over the network by allowing centralized control over limiting access to applications.

82. **Claim 31 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Logston, Johnson, and Piskiel, as applied to claim 1 above, in view of Vange et al (U.S. Patent No. 7,020,783, hereinafter Vange).**

83. With respect to claim 31, the combination of Logston, Johnson, and Piskiel disclose the system of claim 1, but does not disclose a denial of service attack.

However, Vange disclose the terminal-side executable sends a challenge (column 8, lines 17-20, *validate source addresses*) to any third party identified as attempting a denial of service attack on the terminal (column 5, lines 60-63) preventing any additional data traffic to the terminal from the denial of service attack (column 4, lines 12-15, *blocked quickly and/or automatically*).

It would have been obvious to one skilled in the art at the time the invention was made to combine the teachings of Logston, Johnson, and Piskiel with the stability of Vange. The motivation to do so being, to provide a more stable means of

communicating over the network by discontinuing any unwanted network traffic from transmitting.

Response to Arguments

84. Applicant's arguments with respect to claim 1-37 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

85. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Danneels et al Patent No. 5,754,765

b. Bowman-Amuah Patent No. 6,289,382

c. Guheen et al Patent No. 6,473,794

d. Hightower et al Patent No. 6,510,550

e. Goodman et al Patent No. 7,020,697

86. Any inquiry concerning this communication or earlier communications from the examiner should be directed to BLAKE RUBIN whose telephone number is (571) 270-3802. The examiner can normally be reached on M-R: 8:00-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on (571) 272-4001. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

3/31/09

/Rubin Blake/
Examiner, Art Unit 2457

/ARIO ETIENNE/
Supervisory Patent Examiner, Art Unit 2457